

# OSS License Compliance for Software Developers **NOT** for Legal Experts

---

Embedded Linux Conference 2018

March 14

Portland Oregon, US

Satoru Ueda (Satoru.Ueda@sony.com)

Sony Corporation

## About myself

---

- **OSS Community liaison**

- Establishment and fostering the CE Linux Forum since 2003
  - CE Linux Forum: former organization of The Linux Foundation CE Workgroup
- Promote Linux for use in Consumer Electronics appliances

- **Within Sony**

- Open Source Strategy
  - <https://github.com/Sony>
- Reinforcement of OSS License compliance
  - <http://oss.sony.net/Products/Linux>



# The Cathedral and Bazaar ?

---

By whom are you supported in your use of OSS in your company?

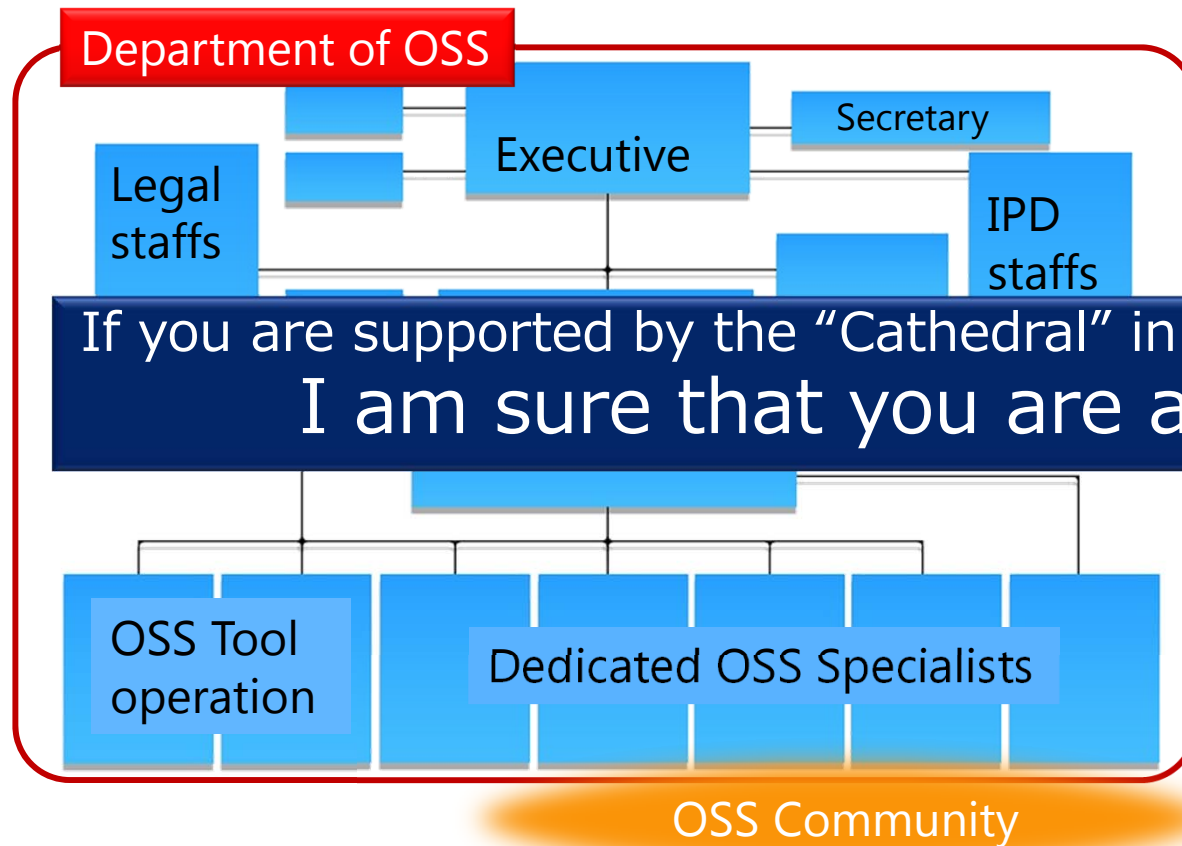


# Cathedral



Department of Open Source Software

## Cathedral



The "Cathedral" take care everything about OSS

If you are supported by the "Cathedral" in every things about OSS  
I am sure that you are a lucky guy!



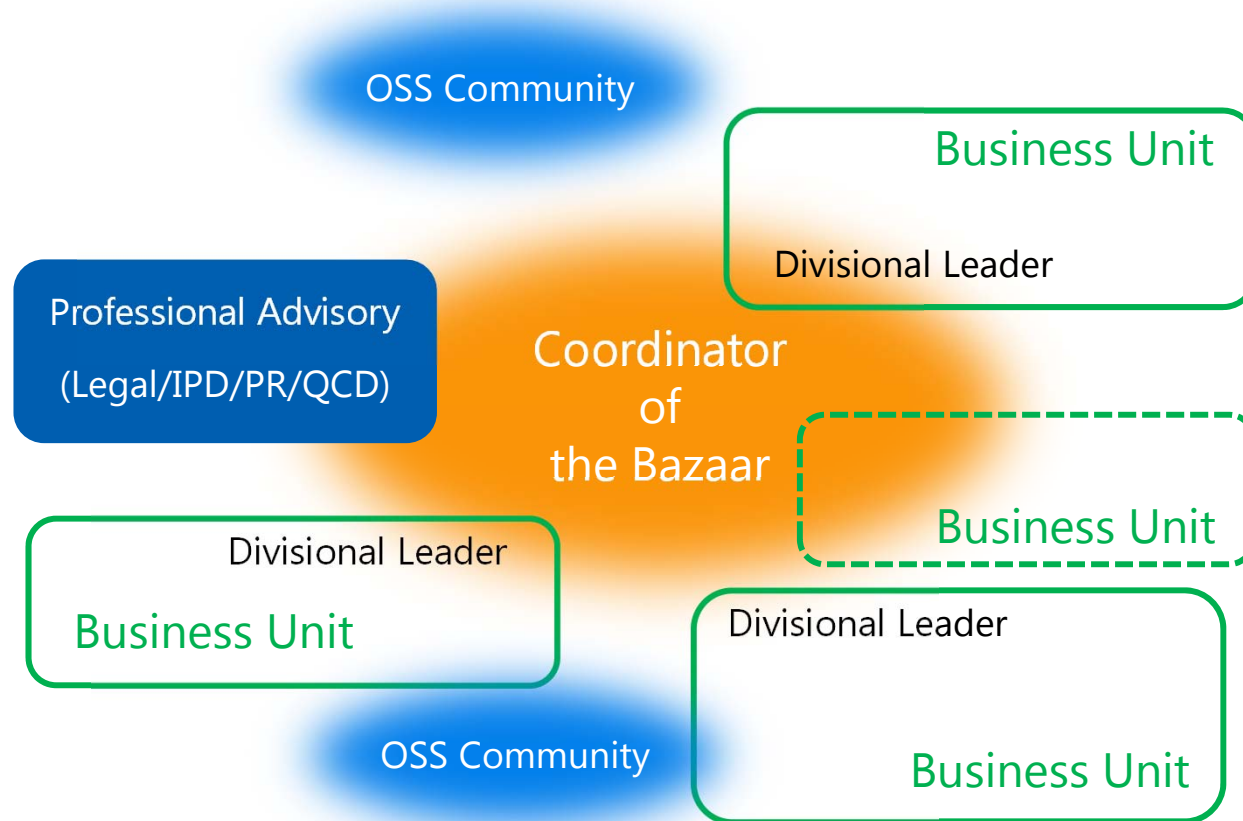


# Bazaar



In-house Open Source Software Community

# In-house Bazaar

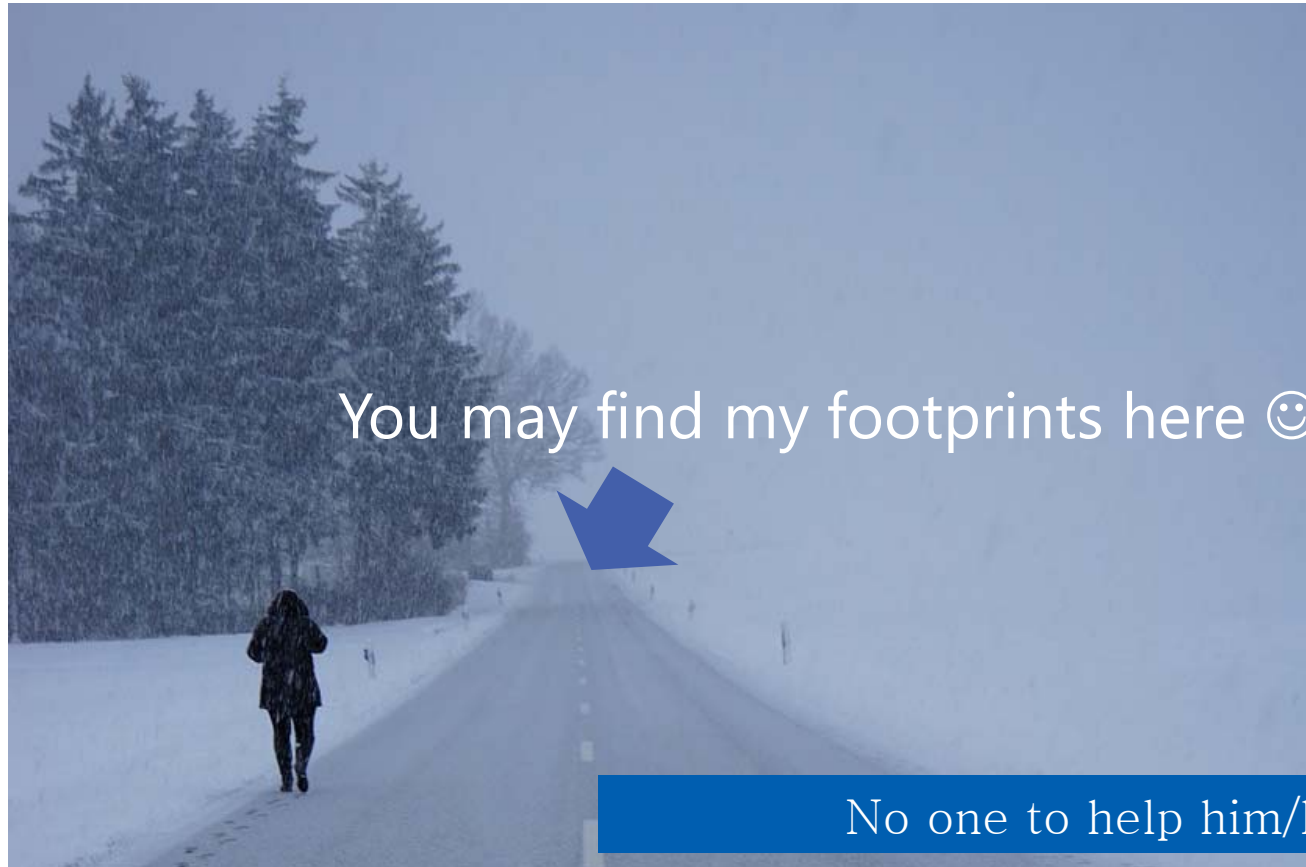


Loose guidelines

Mutual help of the bazaar participants

Each Business Unit takes all responsibility when using OSS

## Lonely Software developer



No one to help him/her who relates with OSS

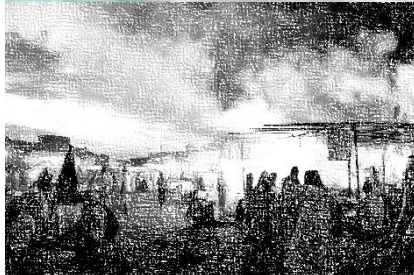


## Ask yourself! Cathedral?

---



Yes I am supported by a great OSS department of my company



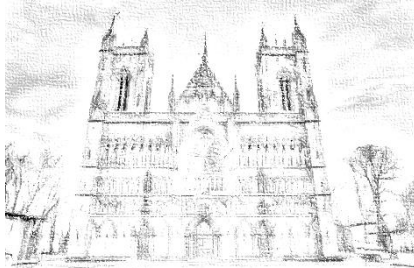
I am not supported by a great OSS department of my company. My company doesn't have such organization. But I belong to an in-house OSS user community which helps each other.



Great organization? In-house community? I am not living in such warm place. I am struggling solely by myself to deal with OSS.

## Ask yourself! Bazaar?

---



Yes I am supported by a great OSS department of my company



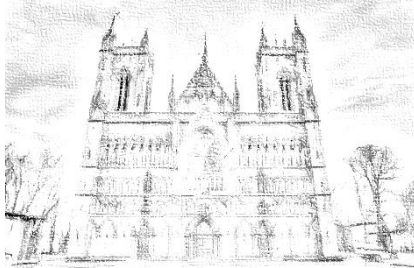
I am not supported by a great OSS department of my company. My company doesn't have such organization. But I belong to an in-house OSS user community which helps each other.



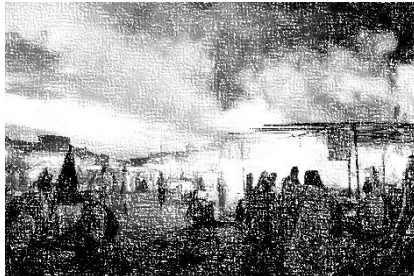
Great organization? In-house community? I am not living in such warm place. I am struggling solely by myself to deal with OSS.

## Ask yourself! Lonely Software Developer?

---



Yes I am supported by a great OSS department of my company



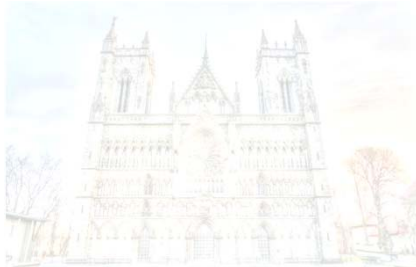
I am not supported by a great OSS department of my company. My company doesn't have such organization. But I belong to an in-house OSS user community which helps each other.



Great organization? In-house community? I am not living in such warm place. I am struggling solely by myself to deal with OSS.

The majority will be: (in the field of embedded system)

---



Yes I am supported by a great OSS department of my company



I am not supported by a great OSS department of my company. My company doesn't have such organization. But I belong to an *in-house OSS user community* which helps each other.



Great organization? In-house community? I am not living in such warm place. I am *struggling solely by myself to deal with OSS.*



# Chapter 1

## If you are the lonely software developer

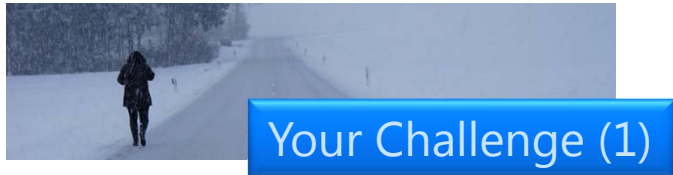
---

- Things which all software engineers should keep in mind.
- How to get other people supported?
- Challenge to create in-house Bazar.



## Avoid two major Misunderstandings of Open Source Software

---



OSS is *unconditional, free-of-charge software*

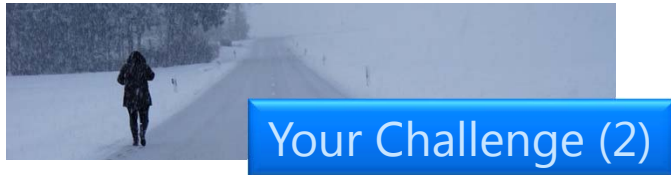
- This misunderstanding may **trap you in a fatal problem**.
- In fact, there was a case, where a company was sued and almost lost.

OSS is *fraught with danger* when it is used

- Developing software without OSS is unrealistic.
- For example, many mobile apps use OSS.
- Knowledge of appropriate use of OSS is necessary.

# Never attempt to hide the use of OSS

---



- **NEVER give up to convince your manager to use OSS**
- **Tell your reason to use OSS**
  - Increase the quality
  - Avoid “reinvention of wheel”
  - Stay at the cutting edge of the software innovation
  - Chances to collaborate with the world famous experts
- **Not recommend to tell “OSS is Free Of Charge software and it will reduce the cost”**
  - For many managers, it is so attractive and irresistible term
  - It may cause serious side effect later
  - As you may know, in order to use OSS you should bear some cost, because every OSS license requires the user to take all responsibility when using OSS

## Find your friends and support mutually (external)

---



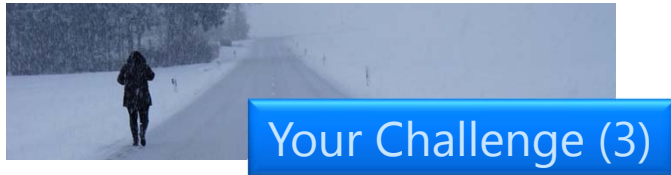
A hint to you

- **Know the practices of other companies**
- **Maybe someone will help you**
  - You will have some chance to help someone
- **Find opportunities to attend OSS events**
  - Embedded Linux Conference
  - Japan Technical Jamboree
  - Initiating your event is also an option



## Learn about the OSS License

---



- **Never give up learning the essence of the OSS License**
  - If you can find someone from Legal expertise, you will be lucky!
- **Understand the generic situation of the legal staffs**
  - Even if you have some legal advisory, in many cases, those legal staffs are not likely familiar with OSS technology, Community and OSS license.

Is this how you think of legal and IPD people?



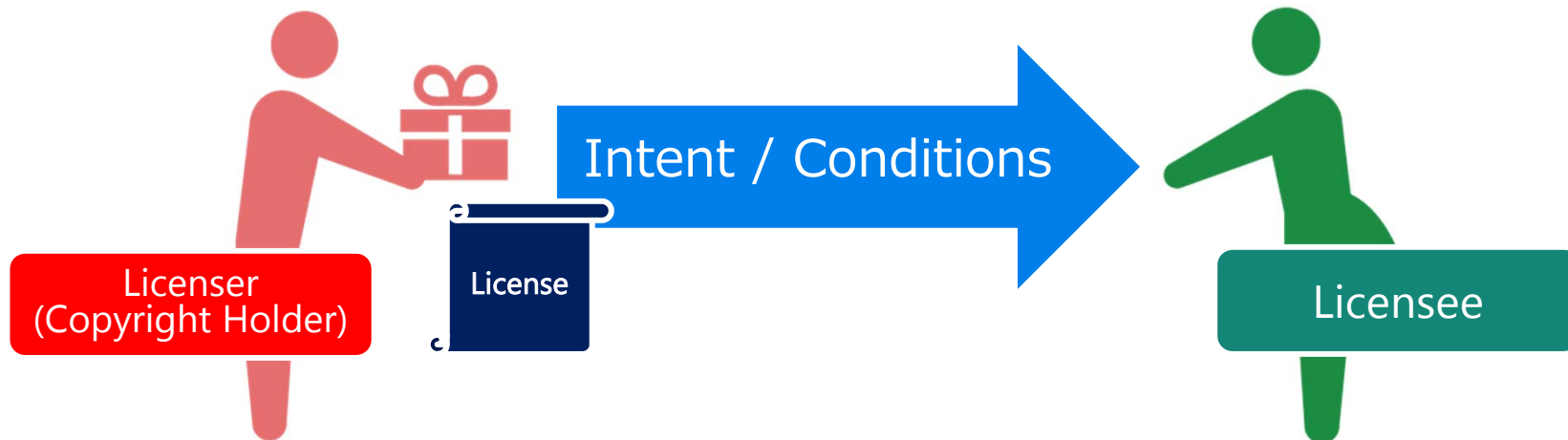
# Ordinal Software License Agreement

From my presentation at *Open Compliance Summit 2016*

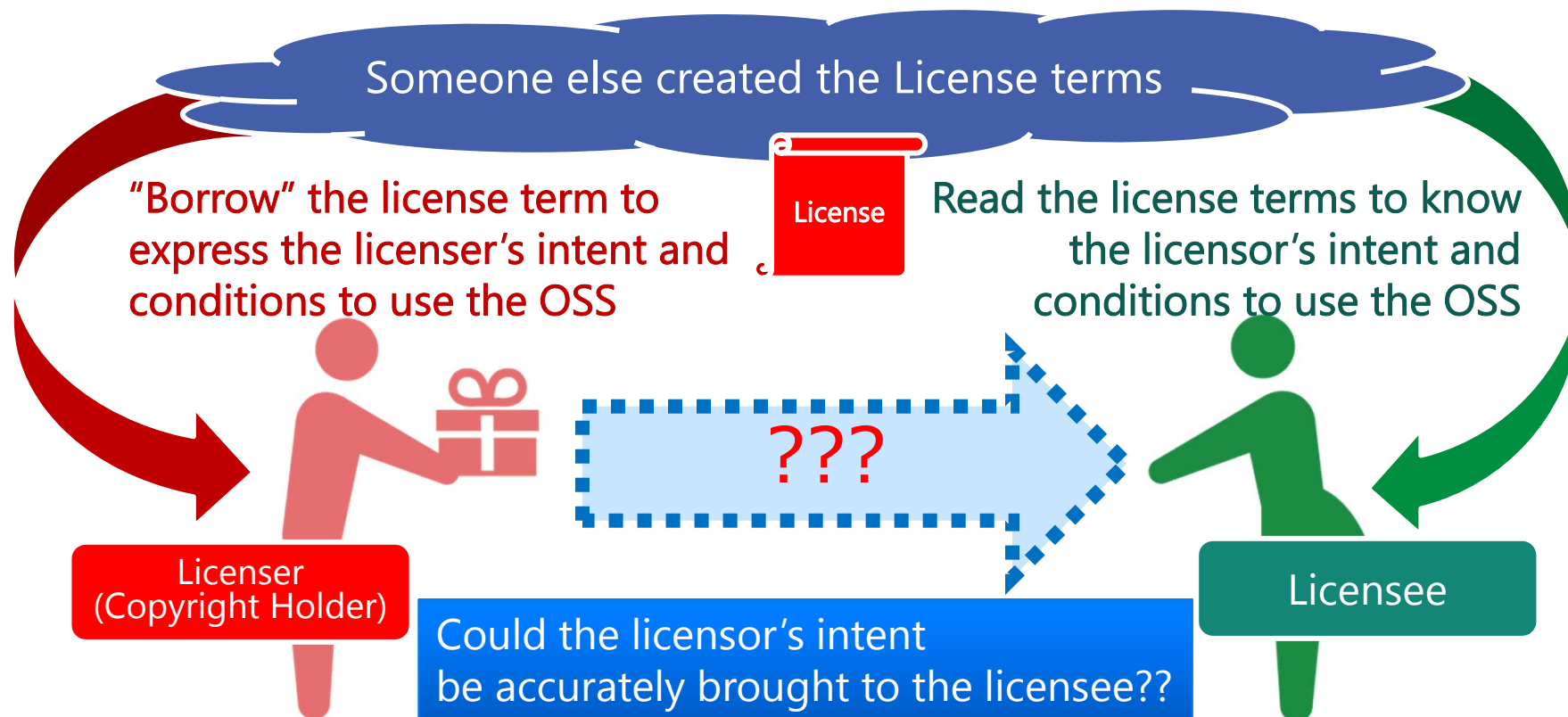
The licenser prepares the license agreement which includes:

- The intent of the licenser to distribute the software
- The conditions to use the software

Through simply reading the license agreement, or the communication with the licenser, the licensee surely understands the condition of the usage and the intent of licenser.

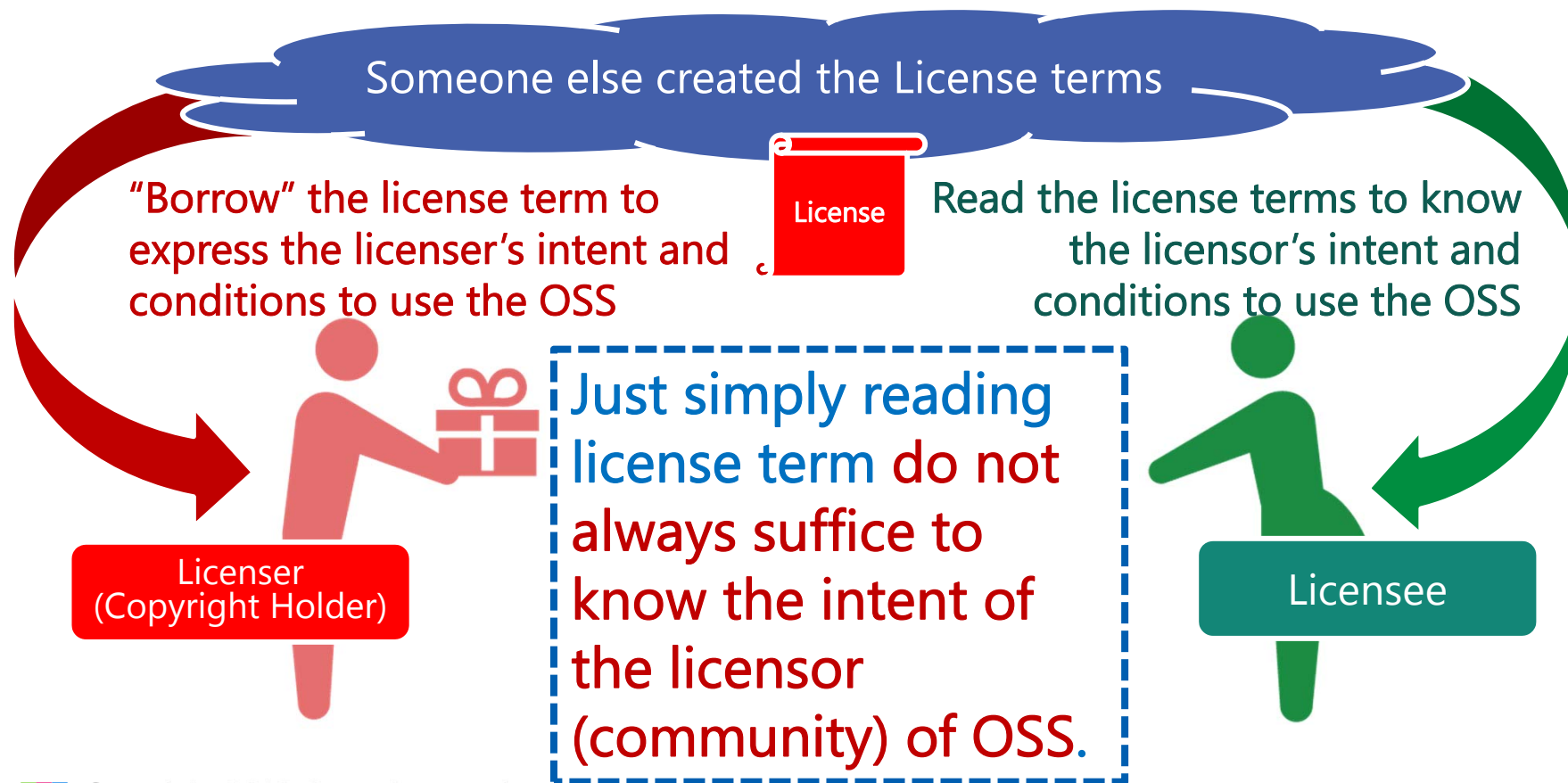


## Trap of OSS License for Legal Experts





## Pit-hole of OSS License for Legal Experts



# Work together!

---

Legal and intellectual  
property department staff

S/W developers who know the  
community and technology

Decision to use OSS &  
Promotion of proper usage

Formation of in-house community

## How can we help his/her learning OSS License?

---

- **My suggestion:**

- **Point out important references to understand OSS License**

- Help “lonely software developer” make their own appropriate rationale for using OSS

- **Examples:**

1. “What is GPL Contamination?”
2. “Why we are not required to publish the source of our program which is dynamically linked to LGPL licensed library?”



## Example (1)

---

**Hints to understand “What is GPL contamination?”**



# Quote from Section 2 of GPL version 2

---

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

Legal staffs: 😊 😊 😊

Good at reading wordy document

However they may hardly understand technical issue

Software developers: 😞 😞 😞

It is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

## What is the Section 2 of GPL version 2 Getting at?

2. (Omitted)

- a) (Omitted)
- b) **You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.**
- c) (Omitted)

These requirements apply to the modified work as a whole. (Omitted) But **when you distribute the same section as part of a whole which is a work based on the Program, the dis**

Software developers: ☹ ☹ ☹

- What defines the “Derivative Work”?

## What is the Section 2 of GPL version 2 Getting at?

2. (Omitted)

a) (Omitted)

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, **to be licensed as a whole at no charge to all third parties under the terms of this License.**

c) (Omitted)

(Omitted) **If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this**  
them as

Software developers: ☹️ ☹️ ☹️

- What does this exactly mean? → **“work are not derived from the Program, and can be reasonably considered independent and separate works in themselves”**

# Let's confirm the FSF FAQs

Where's the line between two separate programs, and one program with two parts? This is a legal question, which ultimately **judges will decide**. We believe that a proper criterion depends both on the mechanism of communication (exec, pipes, rpc, function calls within a shared address space, etc.) and the semantics of the communication (what kinds of information are interchanged). If the modules are included in the same executable file, they are definitely combined in one program.

**Legal staffs:** ☹️ ☹️ ☹️ **shared address space, that almost**

By contrast, **pipes, sockets and command-line arguments are communication mechanisms normally used between two separate programs**. So when they are used for communication, the modules normally are separate programs. But **if the semantics of the communication are intimate enough, exchanging complex internal data structures, that too could be a basis to consider the two parts as combined into a larger program**.

**Software developers:** 😊 😊 😊 **areAggregation**

## Example (2)

---

### **Hints to understand**

**“Why we are not required to publish the source of our program which is dynamically linked to LGPL licensed library?”**

# Read section 6 of the LGPL version 2.1

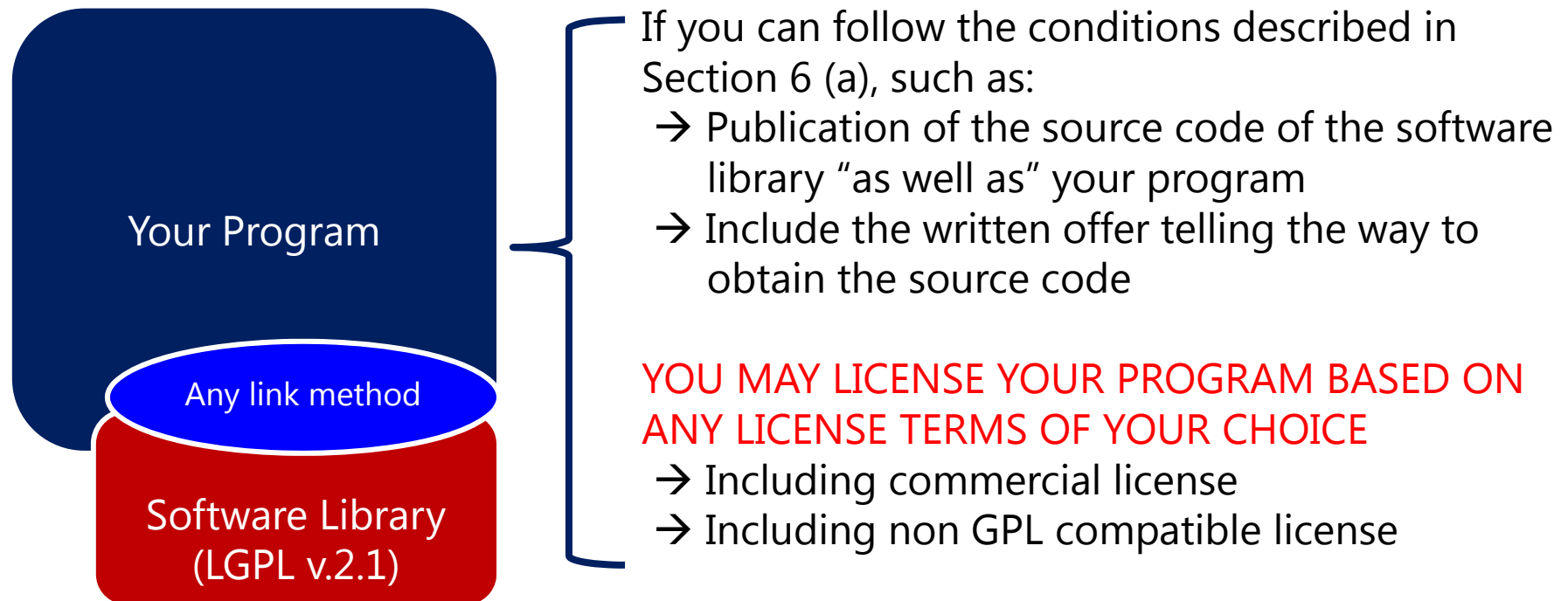
---

- **6.** As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, **provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications. (← YOU MUST NOT FORGET)**
- You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:
  - **a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)**
  - **b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.**
  - **c) d) e) (snip).**
- For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.
- It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.



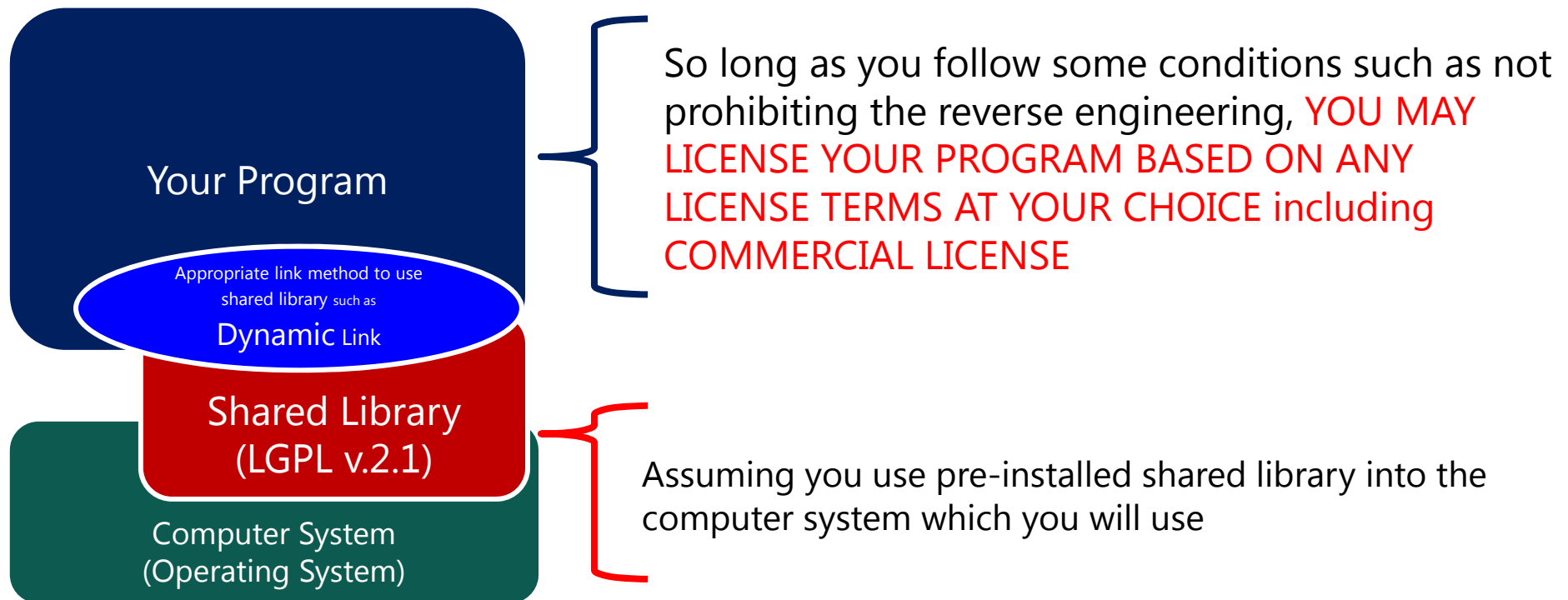
Then you may be able to draw a diagram (1)

Pursuant to LGPL version 2.1 Section 6 (a)



Then you may be able to draw a diagram (2)

Pursuant to LGPL version 2.1 Section 6 (b)



# Toward the next phase

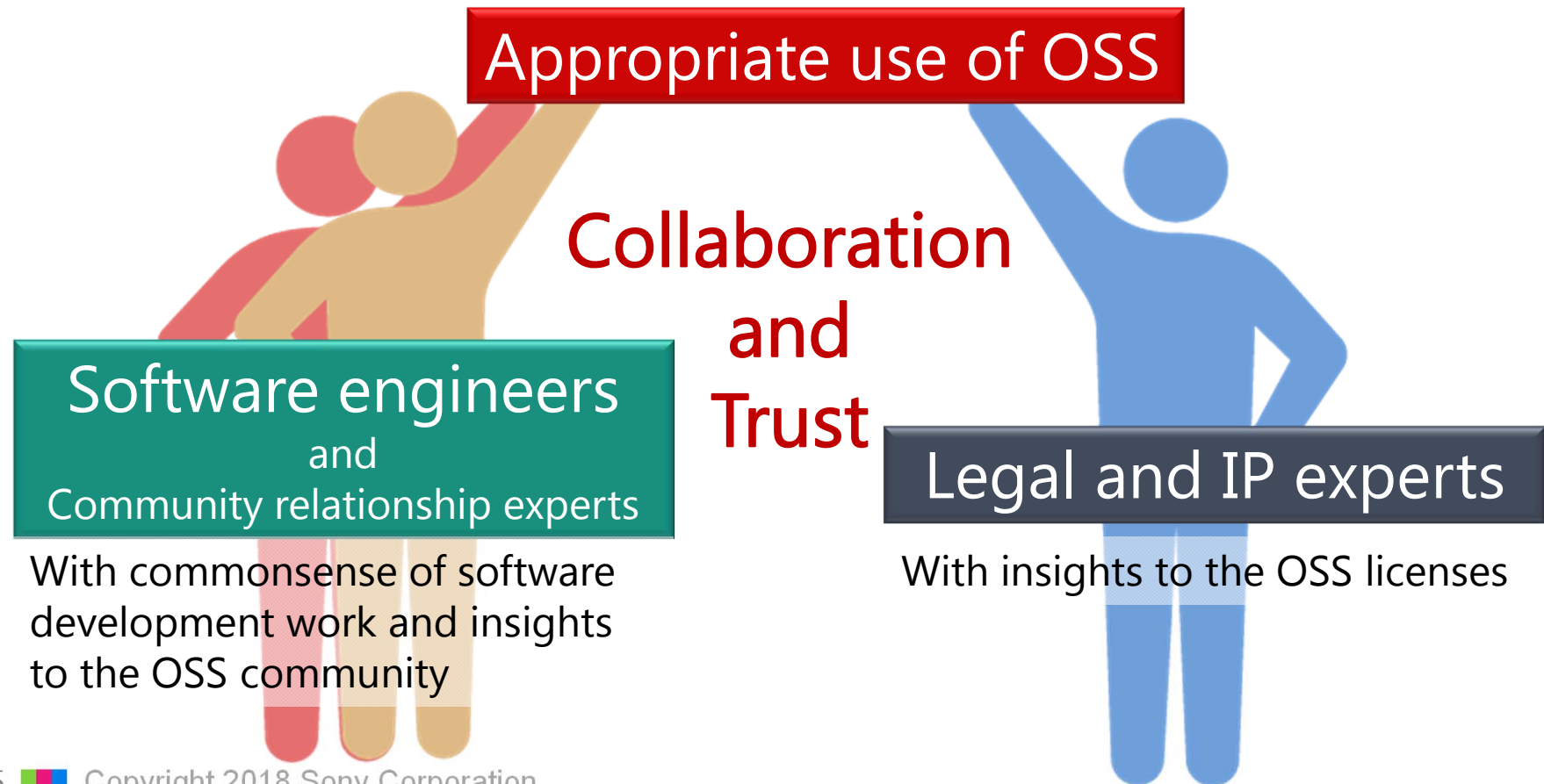
---

## Toward the next step



“Collaboration” is the key word

---



## Chapter 2

### Create and operate in-house Bazaar

---



- Welcome the new people to join the Bazaar
- Key items, guide line, education...
- Is it the goal to create a great cathedral of an OSS department?



## Requirements to establish in-house community

---

- **Talents**

- Software engineers who wish to use OSS
- Open Community relation expert, if one exists
- Legal and IP expert

- **In-House Community Leader**

- **Support from management**

- **Communication tool**

- **All participants**

- Minimum appropriate understandings to OSS
  - Community ecosystem, OSS License, corporate policy and so on

# Sony's current situation



- About 100 members of in-house OSS user community
- OSS strategy board as the community leader (with a few experts)

# Widely open the door to new people to join the Bazaar

---



For the in-house Community members



- **Do not create your own in-house silo of OSS**

- Your in-house bazaar must be open to everyone in your company

- **Get ready for new members to join, at any time**

- Create common rule (guide line to join the bazaar)
- Educate the participants

# In-house Guidelines about OSS



For the in-house Community members

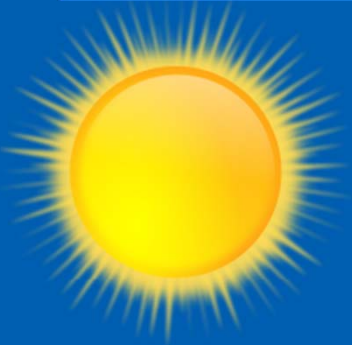
- **“Promote using OSS actively” is the first line**
  - Starting from **“You May Use OSS”**
    - Assuming to comply with the license requirement
    - Assuming at your business unit’s responsibility
  - Very limited idea of **“You Must Not”**
- **Remind what is required by OSS license**
  - License-by-license guidelines for compliance
  - Through collaboration with legal and IPD staffs

Only one “MUST NOT” in Sony’s OSS Guideline

---

**Always keep in mind that**

As a respectful citizen in the community,  
behave honorably.



**Never stray from the right  
path or do something devious!**

Minimum required knowledge for Bazaar participants

Minimum required understandings of Open Source Software, Community and Licenses

Education

All software engineers  
All software related persons  
Who wish to join the Bazaar

In-house Open Source Software Community



## Sony's internal training course material

---



370 Slides  
9 Hours course

Nicknamed: Ueda's OSS Boot Camp



# Open Chain Curriculum may help you



FOSS Training Reference Slides for th

Released under CC0-1.0.

You may use, modify, and share these slides.  
They also come with no warranty.

These slides follow US law. Different legal jurisd  
requirements.

This should be taken into account when using th  
training program.

These slides do not con

## Contents

1. What is Intellectual Property?
2. Introduction to FOSS Licenses
3. Introduction to FOSS Compliance
4. Key Software Concepts for FOSS Review
5. Running a FOSS Review
6. End to End Compliance Management (Example Process)
7. Avoiding Compliance Pitfalls
8. Developer Guidelines

<https://www.openchainproject.org/curriculum>

## Do not stay just a user of OSS



For the in-house Community members

- **Contribution to the community is essential**
  - Do not hoard your bug fix or slight improvement within your company
  - **“Hoarding”** means those technological portion must be maintained by solely yourself which may harmful for you
  - Especially **FORKING is a bad idea**
- **Consider advantage of OSS from view point of technological strategy**
  - In some case making your own technology as OSS and publicize makes great sense

## Advantage of “Bazaar” style

---

- **Flexible, Dynamic and Strategic use of OSS**
- **Adoptability to the diversified use cases of OSS**
  - Consider Sony’s case
- **Foster human resources**
  - “Open collaboration” is a great asset of contemporary software engineers seeking for innovation

## Disadvantage of “Bazaar” style

---

- **Less possibilities to join Open Community**

- Participants of the Bazaar are always too busy for their daily work to contribute to the Open Community
- ➔ I am optimistic because along with the maturity of in-house community, someone may start such activities

- **Efforts to maintain the participants' knowledge and experiences of OSS**

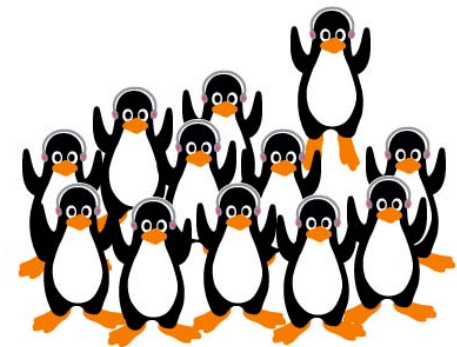
- Community leader's burden for the efforts for education
- ➔ I am thinking about the possibility of Open Collaboration (i.e. Open Chain Project)

- **Less budget**

- ➔ Sorry but no answer ☹

My current question...

Is it my next goal?



Thank you very much  
for your attention!